5th International Workshop
on
TIME-VARYING IMAGE PROCESSING AND MOVING OBJECT RECOGNITION
September 5-6, 1996
Hotel Michelangelo - Florence, Italy

Firenze, April 24 1996

Prof. B.G. Mertzios
Automatic Control Systems Lab.
Department of Electrical and
Computer Engineering
Democritus University of Thrace
Xanthi 67100, Hellas
Greece

ear Prof. Mertzios,

We are very glad to inform you that your paper "Image Block Representation and Its Applications to Manufacturing and Automation" has been accepted for presentation at the 5th International Workshop on "Time-Varying Image Processing and Moving Object Recognition" to be held in Florence, September 5-6, 1996. About 50 papers have been accepted. You will receive by the end of May the final Program. You will receive also by ELSEVIER the sheets and instructions to prepare the final manuscript.

To can present the paper at the Workshop and to have the paper printed in the Proceedings (after the Workshop), you must register yourself. A Registration Form-A and a Reservation Form-B are enclosed. Please check the deadlines for registration and reservation.

Waiting for meeting you at the Workshop, my best regards.

Sincerely yours,

Prof. Vito Cappellini
Workshop Chairman

encl.: 2

Image Block Representation and Its Applications to Manufacturing and Automation

Iraklis M. Spiliotis, Basil G. Mertzios and Nikos Papamarkos

Department of Electrical and Computer Engineering
Democritus University of Thrace
Xanthi 67100
HELLAS

Abstract - **In this paper a vision system for learning and recognition of image objects, with the ability to operate as a real-time industrial vision system for manufacturing and automation tasks is presented. A binary image representation, which is called Image Block Representation and represents the image as a set of nonoverlapping rectangular areas, is applied for the fast and efficient feature extraction on binary images. The block represented binary image is well suited for the fast implementation of various processing and analysis algorithms in a digital computing machine. Specifically, the real-time computation of moments and the efficient structural object description are achieved. The final classifier uses the Dempster-Shafer theory for the combination of the information from these different sources.**

I. INTRODUCTION

For the learning and recognition of the image objects binary images are usually used, since only the information of the shape is important for these tasks. The most common image representation format is a two-dimensional (2-D) array, each element of which has the brightness value of the corresponding pixel. For a binary image these values are 0 or 1. In a serial machine, only one pixel is to be processed at a time, by using the 2-D array representation. However, many image processing and analysis operations, may be executed in parallel using a parallel machine. Many research efforts considered the problem of selecting an image representation suitable for concurrent processing in a serial machine. The need for such approaches arises from the fact that an image contains a great amount of information, thus rendering the processing a difficult and slow task. Existing approaches to image representation aim to provide machine perception of images in pieces larger than a pixel and are separated in two categories: Boundary based methods and region based methods. Such approaches include quadtree representations [1], chain code representations [2], contour control point models [3], autoregressive models [4], the interval coding representation [5] and block implementation techniques [6]-[8]. One common objective of the above methods is the representation of an image in a more suitable form for a specific operation. Recently a new advantageous representation for binary images, which is called *image block representation* (IBR) has been presented [9][10]. In the block representation process the whole binary image is decomposed in a set of rectangular areas with object level. The image block representation exploits the fact that many compact areas of a given binary image have the same value. This representation constitutes an efficient tool for image processing and analysis techniques.

Various sets of 2-D statistical moments constitute a well-known image analysis and pattern recognition tool. Moments have been used for various image processing and analysis tasks, including shape analysis, scene matching, image normalization, object and character recognition, and three-dimensional object recognition [11]-[20]. In pattern recognition applications, a small set of the lower order moments is used to discriminate among different patterns. The most common moments are the geometrical moments, the central moments, the normalized central moments and the moments invariants [17], [18]. Other sets of moments are the Zernike moments and the Legendre moments (which are based on the theory of orthogonal polynomials) [19], [21], and the complex moments [20].

One main difficulty concerning the use of moments as features in image analysis applications is the implied high computational time. A number of approaches that reduce the computational time concerning calculation of moments have been appeared [3], [23]-[26]. In [23], [24] and [25], the problem has been reduced from two-dimensional to a one-dimensional one using Green's theorem; this approach reduces the complexity from $O(N^2)$ to $O(N)$, since only the boundary pixels are considered and the length P of the boundary is linearly related to $\sqrt{A}$, where A is the object area. This approach results to a significant computational gain but it is inferior to the block based computation since it is dependent to the object boundary. In [3] control point models based on the least-square normalized B-splines, are used for the representation of the object boundary and the complexity of the moments computation is analogous to the shape model order and independent of the scale. In [26] the computation formula of its one central moment has been considered as an impulse response of a filter, which is then transformed to the z-domain and the transfer function of the corresponding digital filter is obtained. The computational complexity for the calculation of the 16 central moments up to the order (4,4) of an image with NxN points, is $4N^2+16N+80$ additions and only 32 multiplications or power calculations. This approach is also inferior to the block based computation since is dependent on the image size. Using the block representation in binary images, real-time computation of 2-D statistical moments is achieved through analytical formulae for the moment computation of each block of the image. The computational complexity of the proposed technique is $O(L^2)$, where *(L-1, L-1)* is the order of the 2-D moments to be computed [9].

In many computer vision systems which find application in manufacturing and automation processes, thinning constitutes a basic and significant step in the recognition process. The skeleton of an object provides useful information about the shape of the object. In addition this information may be considered as independent of the noise and of the width of the object. A thinned pattern is a line drawing representation of a usually elongated pattern. A number of approaches for thinning have been proposed in the past. The Medial Axis Transform (MAT) introduced by Blum [27] is a classical way to obtain skeleton. Thinning methodologies are described in [28]-[34]. Using image block representation, the structural description of the object using a fast estimation of the skeleton is achieved. The image block representation method results to a set of points and all the necessary information concerning the links among them; therefore a representation of the skeleton is obtained. Moreover the endpoints and the treepoints are provided by the above algorithm. These critical points are necessary for the structural description of the object; since a set of subpatterns is formed from these points. The extracted subpatterns are classified, using simple geometrical and statistical features. The structural description of the object is obtained by the classifier and is used for object learning and/or recognition.

The final classifier combines the information from these two different sources (moment invariants and the structural description) and makes a decision about the classification of the input object in a certain pattern class. The Dempster-Shafer theory of evidential reasoning is used in the final classifier.

II. IMAGE BLOCK REPRESENTATION

A bilevel digital image is represented by a binary 2-D array. Without loss of generality, we suppose that the object pixels are assigned to level 1 and the background pixels to level 0. Due to this kind of representation, there are rectangular areas of object value 1, in each image. These rectangulars, which are called, have their edges parallel to the image axes and contain an integer number of image pixels. At the extreme case, the minimum rectangular area of the image is one pixel.

Consider a set that contains as members all the nonoverlapping blocks of a specific binary image, in such a way that no other block can be extracted from the image (or equivalently each pixel with object level belongs to only one block). It is always feasible to represent a binary image with a set of all the nonoverlapping blocks with object level and this representation is called *Image Block Representation* (IBR). According to the above discussion, two useful definitions concerning IBR are formulated:

Definition 1

Block is called a rectangular area of the image, with edges parallel to the image axes, that contains pixels of the same value.

■

Definition 2

A binary image is called *block represented,* if it is represented as a set of blocks with object level, and if each pixel of the image with object value belongs to one and only one block.

■

According to Definitions 1 and 2, it is concluded that the IBR is an information lossless representation. Given a specific binary image, different sets of different blocks can be formed. Actually, the nonunique block representation does not have any implications on the implementation of any operation on a block represented image.

The IBR concept leads to a simple and fast algorithm, which requires just one pass of the image and simple bookkeeping process. In fact, considering a $N_1 \mathrm{x} N_2$ binary image *f(x,y),* $x=0,1, \ldots, N_1-1$, $y=0,1, \ldots, N_2-1$, the block extraction process requires a pass from each line *y* of the image. In this pass all object level intervals are extracted and compared with the previous extracted blocks. In the following, an IBR algorithm is given.

Algorithm 1: *Image Block Representation.*

Step 1: Consider each line *y* of the image f and find the object level intervals in line *y*.

Step 2: Compare intervals and blocks that have pixels in line *y-1*.

Step 3: If an interval does not match with any block, this is the beginning of a new block.

Step 4: If a block matches with an interval, the end of the block is in the line *y*.

■

As a result of the application of the above algorithm, we obtain a set of all the rectangular areas with level 1 that form the object. A block represented image is denoted as:

$$f(x, y) = \{b_i : i = 0,1,\ldots,k-1\} \quad (1)$$

where k is the number of the blocks. Each block is described by four integers, the coordinates of the upper left and down right corner in vertical and horizontal axes. The block extraction process is implemented easily with low computational complexity, since it is a pixel checking process without numerical operations. Fig. 1, illustrates the blocks that represent an image of the character d.

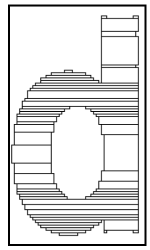


Figure 1. Image of the character d and the blocks.

III. COMPUTATION OF MOMENTS

Consider a binary digital image $f(x,y)$, with N_1 pixels in horizontal axis and N_2 pixels in vertical axis. The 2-D geometrical moments of order (p,q) of the image are defined by the relation:

$$m_{pq} = \sum_{x=0}^{N_1-1} \sum_{y=0}^{N_2-1} x^p y^q f(x,y), \quad p,q = 0,1,2,... \tag{2}$$

Since the background level is 0, only the pixels with level 1 are taken into account for the computation of the moments. Thus, the 2-D geometrical moments of order (p,q) of the image $f(x,y)$ are defined by the relation:

$$m_{pq} = \sum_{x} \sum_{y} x^p y^q \quad \forall x, y: f(x,y) = 1 \tag{3}$$

Specifically, if the image $f(x,y)$ is represented by k blocks, as it is described in (1), all the image pixels with level 1 belong to the k image blocks and therefore (3) may be rewritten as:

$$m_{pq} = \sum_{i=0}^{k-1} m_{pq}^{b_i} = \sum_{i=0}^{k-1} \sum_{x=x_{1,b_i}}^{x_{2,b_i}} \sum_{y=y_{1,b_i}}^{y_{2,b_i}} x^p y^q \tag{4}$$

where x_{1,b_i}, x_{2,b_i} and y_{1,b_i}, y_{2,b_i} are the coordinates of the block b_i with respect to the horizontal axis and to the vertical axis, respectively.

In (4), if the rectangular form appeared within the blocks is taken into account, then the geometrical moments of one block b, with coordinates x_{1b}, x_{2b}, y_{1b}, y_{2b}, are given by

$$m_{pq}^{b} = \sum_{x=x_{1b}}^{x_{2b}} \sum_{y=y_{1b}}^{y_{2b}} x^p y^q = x_{1b}^p \sum_{y=y_{1b}}^{y_{2b}} y^q + (x_{1b}+1)^p \sum_{y=y_{1b}}^{y_{2b}} y^q + \ldots + x_{2b}^p \sum_{y=y_{1b}}^{y_{2b}} y^q = \left(\sum_{x=x_{1b}}^{x_{2b}} x^p \right) \left(\sum_{y=y_{1b}}^{y_{2b}} y^q \right) \tag{5}$$

Using the rectangular form appeared within the block, the computational effort, which is characterized by the complexity $O(N^2)$ for the calculation of moments using (2), is reduced to $O(N)$ for the calculation of moments using (5). For the computation of (5), it is adequate to calculate the following summations of the powers of x and y :

$$S_{x_{1b},x_{2b}}^{p} = \sum_{x=x_{1b}}^{x_{2b}} x^p, \quad S_{y_{1b},y_{2b}}^{q} = \sum_{y=y_{1b}}^{y_{2b}} y^q, \quad x, y, p, q \in Z \tag{6}$$

Moreover, taking into account the known formulae:

$$S_{1,n}^{1}=\frac{n(n+1)}{2}\ ,\ S_{1,n}^{2}=\frac{n(n+1)(2n+1)}{6}\ ,\ S_{1,n}^{3}=\frac{n^{2}(n+1)^{2}}{4}\ ,\ S_{1,n}^{4}=\frac{n(n+1)(2n+1)(3n^{2}+3n+1)}{30} \tag{7}$$

$$\binom{m+1}{1}S_{1,n}^{1}+\binom{m+1}{2}S_{1,n}^{2}+\ldots+\binom{m+1}{m}S_{1,n}^{m}=(n+1)^{m+1}-(n+1)\ ,\ \ \forall m\in Z^{+}$$

and finally

$$S_{x_{1b},x_{2b}}^{p}=S_{x_{2b}}^{p}-S_{x_{1b}-1}^{p} \tag{8}$$

Extending the above presented analytical method the real-time computation of the central moments, the normalized central moments and the moment invariants is also achieved [9].

IV. STRUCTURAL PATTERN RECOGNITION

An object normalization procedure is first executed in order to preserve rotation invariant descriptions of the objects. Specifically the maximal axis of the object is found and the whole object is rotated in such a way that the maximal axis has a vertical position and that the upper half of the image object contains the most of the object's maze.

A. Thinning

Thinning algorithms should preserve topological and geometrical properties, compress the image data and have high processing speed. One problem concerning the preservation of geometrical properties is that of locality: The small local neighborhoods that are used for simplicity in the various algorithms, are incapable to provide global information and in particular to distinguish between noise and end points. This also may lead to excessive erosion of the object. Finally, the measure of the quality of a given thinned pattern is subjective and dependent on the application.

In this paper a fast noniterative thinning method for block represented binary images is presented. The method has low computational complexity, extracts only critical points and to a degree appears to have immunity to the problem of locality. The thinning procedure is achieved by extracting a set of critical points of the object and properly connecting them. For the description of the thinning algorithm, the following Definitions are used [44]:

Definition 3

Group is an ordered set of connected blocks, in such a way that all its intermediate blocks are connected with two other blocks, while the first and last blocks are connected with only one block.

■

Definition 4

Junction point is called a point that it is connected with two other points and it is an intermediate point of the thinned pattern.

■

Definition 5

End point is called a point that it is connected with only one other point and it is a point of termination of a thinned pattern.

■

Definition 6

Tree point is called the point that it is connected with more than two other points and it is a point where a thinned pattern is separated to three or more parts.

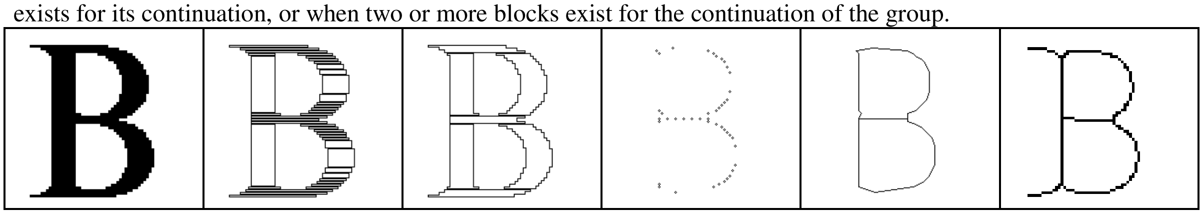
■

Definition 7

Critical point is called a junction or an end or a tree point.

■

The locality problem is addressed in the algorithm. This is achieved by the use of a suitable neighborhood at each case. Specifically, groups of connected blocks are formed. Each group is terminated when an adjacent block does not exists for its continuation, or when two or more blocks exist for the continuation of the group.



(a)	(b)	(c)	(d)	(e)	(f)

Figure 2. (a). Image of the character B. (b) The extracted blocks. (c) The groups of blocks. (d) The critical points. (e) The links among the critical points results to the thinned characters. (f) The resulted thinned characters using an iterative algorithm.

Each group defines a local neighborhood and all the necessary processing takes place in this neighborhood. Using a few simple rules for the processing, the groups are checked and labeled to certain categories:

- *Vertical Elongated groups*. The absolute value of the angle of these groups with the horizontal axis is usually greater than $30°$. The width of each block of a vertical elongated group should not exceed a threshold value. The connections among the blocks result to junction points, which belong to the thinned line that results from the group. For each pair of connected blocks, one junction point (the central point of the common line segment of two connected blocks) is extracted. For each block we check if the distance among its junction points and its extremities (i.e. the central points of the edges of the small dimension) of the block, exceeds a threshold value. In such a case, in the corresponding extremity of the block, an end point of the skeleton is extracted, which belongs to the block.
- *Horizontal Elongated groups*. The absolute value of the angle of these groups with the horizontal axis is smaller than $30°$. The width of an horizontal elongated group is significantly greater than its height and also its height appears to have small variation. Due to the horizontal scanline of the IBR Algorithm, the connections among the blocks in these horizontal oriented groups result to junction points that belong to a small vertical line at the middle of the group, which obviously is not accepted as the thinning of the neighborhood. For the extraction of the junction points the algorithm starts from the left end of an horizontal elongated group and moves to the right with constant width steps. At each step a junction point is extracted at the middle of the height of the group at this vertical position.
- *Angle groups*. The angle groups are connected with two other groups that lie on the same vertical or horizontal side of the angle group. The width and the height of an angle group are usually small. An angle group should not be connected to a noisy group. If a group has labeled as angle group and it is connected with a noisy group, then the label "angle" is replaced by the horizontal elongated label or the vertical elongated label. Three junction points are extracted from an angle group. The two junction points are extracted due to the connections with the two groups and another one for the formulation of an angle.
- *Noisy groups*. These are small and spurious branches of the object. The noisy groups have width and height less than a threshold and they are connected to only one group, which is not an angle group. In the most cases, the noisy groups are connected from the left or right side to vertical elongated groups or from the up or down side to horizontal elongated groups. In these cases the extraction of junction points from the noisy groups is not acceptable, according to human perception about thinning; otherwise a noisy end point would be created. The noisy groups are branches of the object that have small height and width and usually they do not contribute to the thinned pattern of the object. Junction points are extracted from the noisy groups, if and only if the noisy group is connected at the ends of an elongated group.

In the case where a block has more than two junction points or end points, appropriate junction points that are located near the middle of the critical points are selected as tree points. If such a central junction point does not exist, then the middle of the critical points is selected as the tree point of the block. The graphical link of the critical points belonging to the same block, forms the image of the skeleton of the object. In the case where a tree point exists in a block, each other critical point of the block is linked only with the tree point.

The above method is fast and is in congruence with the human perception about the thinned pattern of an elongated object. An extension of this method is the formulation of subpatterns that are used for a structural description of the object. A subpattern is defined between each pair of the end or tree points belonging to the same block or belonging to different blocks and are connected through junction points.

Another advantage of the proposed method is that it allows the structural description of elongated objects, as it is presented in the next subsection.

Fig. 2 demonstrates (a) an image of the character B, (b) the extracted blocks, (c) the groups of blocks, (d) the critical points and (e) the resulted thinned characters. Fig. 2 (f) illustrates the resulted thinned character using the algorithm found at [13], [22]. The proposed algorithm is 14 times faster for the 64x64 image of Fig. 12, in comparison with the iterative thinning (point removal) and specifically the algorithm found at [13], [22], as it results from simulations. For larger images the factor of time reduction is much greater.

B. Structural Pattern Recognition

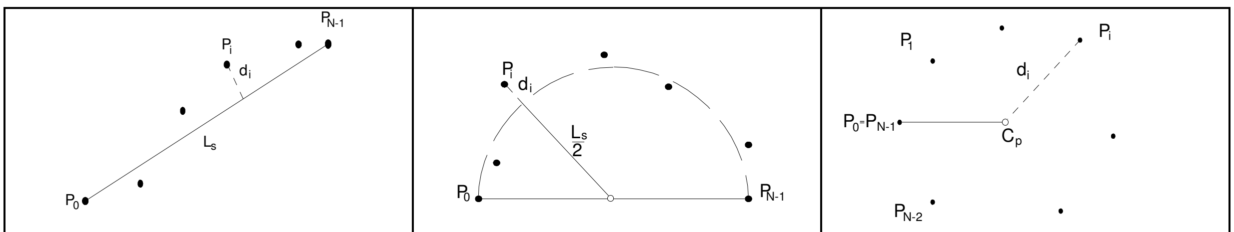
A structural pattern recognition system for elongated objects that operates in two stages is presented here. In the first stage the classification of the subpatterns that constitute the whole object takes place. In the second stage the description and the classification of the whole object is achieved.

A subpattern is a sequence of points in Z^2. The task in the first stage is the classification of the subpatterns in a specified number of the classes. These classes are defined as [10]:

1. *L*, for a straight line segment.
2. *S*, for a semicircle.
3. *C*, for a circle.

We use only these three classes in order to simplify the problem. Each subpattern is classified as one or more of the three basic classes. The classification of each subpattern is based on simple geometrical features. At first, the equation and the length (L_s) of the *Hypothetical Line* (*HL*), which passes from the first P_0 and the last P_{N-1} point of the subpattern is formed. Now the following hold:

- If the deviation of the distances from the intermediate points P_i, i = 1,2,...,*N*-2 to the *HL* is small according to the length L_s of the *HL*, then the subpattern is classified as class *L*.
- If the deviation of the distances from the intermediate points P_i, i = 1,2,...,*N*-2 to the center of the hypothetical circle, which is defined with center the centroid of the *HL* and with diameter equal to the length L_s, is small according to the radius L_s/2, and if the first and the last points of the subpattern are different, then the subpattern is classified as class *S*.
- If the deviation of the distances from the intermediate points P_i, i = 1,2,...,*N*-2 to the center of the hypothetical circle, which is defined with center the centroid of the *HL* and with diameter equal to the length L_s, is small according to the radius L_s/2, and if the first and the last points of the subpattern are the same, then the subpattern is classified as class *C*.
- In every other case, where a subpattern is not classified to any of the above three classes, then the algorithm searches for a suitable point which divides the subpattern to two new subpatterns. The separation may result to subpatterns which belong to any of the desired classes (*L*,*S*,*C*).



(a) (b) (c)

Figure 3. (a) The class-L, (b) the class-S and (c) the class-C.

The above classification procedure is fast implemented. In every step only one feature is examined. In the case where a subpattern is classified as class *L*, then the other features are not calculated. Therefore, the feature extraction process is characterized by low computational complexity, since it is controlled by the classifier and only the necessary features are calculated. The classification of the subpatterns in the three classes *L*, *S*, *C* using the above described criteria is satisfied.

At the second stage, various classifiers may be used for the classification of the whole object. A matching like classifier is presented here. The subpatterns are positioned on a specified grid of dimension *g*x*q* over the image of the object. The choice of the magnitude of the grid (the *g* and *q*) depends on the considered application. Specifically, the grid should satisfy the requirement of local generalization, in order to avoid the small variations due to the noise and to reduce the complexity of the system. On the other hand, we should avoid very large grids, which result to low discrimination efficiency.

Each subpattern is described by its type, its position on the grid and its orientation. The information about the orientation is provided only for class *L* or class *S* subpatterns and is a number between 0 and 7. Information concerning the locations of the connections among the subpatterns is also provided. This is a complete description of the object, or equivalently the reconstruction of a thinned pattern of the object is feasible using this description.

The input object is classified to the class that has the same number and type of the subpatterns located at the same cells according to the *g*x*q* grid, and also the same number and location of connections among the subpatterns.

V. THE FINAL CLASSIFIER

A. Introduction to Dempster-Shafer theory

The mathematical theory of evidence, known as Dempster-Shafer theory, was first developed by Dempster in the 1960s [35]. Subsequently, it was extended by Shafer [36]. Recently, this method has become a promising method for dealing with problems arising in combining evidence and managing uncertainty and ignorance [37]-[41]. The theory deals with the familiar idea of assigning a degree of belief to certain propositions on the basis of combining the available evidence.

In [36] Shafer proposed to split the total amount of knowledge, into quantities assigned to some events for which evidence is available as to whether they are to occur. These elements are called focal elements. A mapping *m* is defined which allocates to each event a portion of the total knowledge:

$$m(\varnothing) = 0 \tag{9}$$

$$\sum_{B \subseteq \grave{U}} m(B) = 1 \tag{10}$$

where *Ω* is the space representing all the possible focal elements and is called frame of discernment. The mapping *m* is called basic probability assignment. It is important to notice that a basic probability can be assigned not only to a single element of *Ω* but also to a subset of *Ω*. The belief or the lower probability assignment of the event *A* is calculated on the basis of events whose occurrence imply that of *A*:

$$\mathrm{Bel}(A) = \sum_{B \subseteq A} m(B) \tag{11}$$

The plausibility or the upper probability assignment of the event A is calculated on the basis of the events *B,* which may occur simultaneously with *A* and is given by:

$$\mathrm{Pl}(A) = \sum_{B \cap A \neq \varnothing} m(B) \tag{12}$$

It can be proved that $\mathrm{Bel}(A) \leq \mathrm{Pl}(A)$ and $\mathrm{Bel}(A) = 1 - \mathrm{Pl}(\overline{A})$. As a reasoning theory, the Dempster's rule of combination provides a method for change prior opinions in the light of new evidence. Suppose are obtained two bodies of evidence from two independent sources and they are expressed by the basic probability assignments m_1 and m_2 . Suppose that the basic probability assignment m_1 of the event *B* has the focal elements B_i , *i*=1,...,*n* and the basic probability assignment m_2 of the event *C* has the focal elements C_j , *j*=1,...,*m*. The combined evidence *A* has the basic probability assignment:

$$m_{1,2}(A) = \frac{\sum_{B_i \cap C_j = A} m_1(B_i) m_2(C_j)}{1 - \sum_{\hat{A}_{\acute{e}} \cap C_j = 0} m_1(\hat{A}_{\acute{e}}) m_2(C_j)} \ , \quad \forall A \neq \varnothing \tag{13}$$

A basic difference with the Bayessian reasoning is that the belief is assigned to the focal elements only if some evidence occurs. Suppose that *Ω* consists from four focal elements *A, B, C, D* and that an evidence confirms the focal element *A* with belief μ. The remaining belief 1- μ is assigned to *Ω*. Thus *m*(*A*)= μ, *m*(*Ω*)=1- μ, *m*(*B*)=*m*(*C*)=*m*(*D*)=0. The idea behind the Dempster-Shafer method is that since no other evidence exists, except *m*(*A*)= μ the remaining *m*(*Ω*) represents ignorance and not disbelief to the elements *B, C, D.* Therefore the remaining *m*(*Ω*) is a belief mass free to be assigned to anyone of the focal elements upon the knowledge of new evidences. Also Bel(*A*)= *m*(*A*)= μ and Pl(*A*)= *m*(*A*)+ *m*(*Ω*)=1, which means that Bel and Pl correspond to upper and lower probability of *A*. In this situation Bayessian reasoning should assign the remaining belief to the negation of *A*, the focal elements *B, C, D*.

Consider now another situation, where there is no evidence for preference of any subset of *Ω*. Bayessian reasoning will assign 0.25 to each focal element in order to represent ignorance. The Dempster-Shafer theory assigns 1 to *Ω* and 0 to every other focal element. The risk with Bayessian reasoning is that there is no difference in representing a situation where each focal element has a belief of degree 0.25.

B. Object Learning and Classification

The final classifier of the vision system uses the Dempster's rule in order to combine the evidences from two different sources of information. In the proposed system the kind of information is statistical as provided by the moment invariants and structural since it is provided by the subpatterns approach.

During the learning process a set of input objects are presented to the system, they analyzed according to the different methods and their class is given as input to the system by a human.

In the following it is explained how the basic probability assignments are acquired for both methods.

Classification based on the moment invariants.

The logarithms of the moment invariants are considered as features for the recognition in order to reduce the dynamic range. In the following the similarity measures among the feature vector $\mathbf{f}_I$ of the input object and the feature vectors $\mathbf{f}_i$ i=1,2,...,N of the known patterns that have been stored in a database, are performed. The measures of similarity are given by:

$$d_{Ii} = d(\mathbf{f}_I, \mathbf{f}_i) = 1 - \frac{|\mathbf{f}_I - \mathbf{f}_i|}{\sqrt{\mathbf{f}_I^2 + \mathbf{f}_i^2}} \ , \quad i = 1,2,...,N$$

In the following these distances d_{Ii} are divided by the max{ d_{Ii} }, therefore they are normalized to the range [0,1] and a possibility distribution [42] is obtained π_i , i=1,2,...,N. These possibility measures are reordered in such a way that the ordered set of their focal elements to be a consonant set, i.e.:

$$\pi_1 \geq \pi_2 \geq \ldots \geq \pi_N \quad \Leftrightarrow \quad A_1 \subseteq A_2 \subseteq \ldots \subseteq A_N$$

where the set A_1 contains the pattern with the maximum similarity measure, the set A_2 contains the elements of the set A_1 and the next pattern with the second largest similarity measure, e.t.a. Therefore, as it has been stated in [42], the relations among basic probability assignments and the possibility measures are obtained by:

$$m(A_i) = \pi_i - \pi_{i+1}$$

for all i, where $\pi_{N+1} = 0$.

Classification based on the structural information

The input object is presented to the structural pattern recognition system and analyzed as described in Section IV. The structural and qualitative nature of the object description does not implies any exact and accurate measurements. Therefore, the procedure of acquiring the basic probability assignments lacks a strong theoretical background and is based on intuitive and experimental criteria.

The procedure of assigning the basic probabilities should satisfy the following requirements:

- To assign the highest basic probability to a specific pattern, if the structural description of the input object is the same with the description of the pattern.
- To assign a high value basic probability to a pattern that has a quite similar description with the input object.
- To assign a low value basic probability to a pattern that has an important different description with the input object.

A procedure that satisfies the above constraints is the following:

The half of the initial maximum basic probability (i.e. 0.5) is assigned to a pattern that has exactly the same type and location of end and tree points as the input object. Each difference at the location and the type of the end and tree points decreases the amount of the initial basic probability that is assigned to the pattern by a certain penalty value. Suppose that the input object has N_I end and tree points, that the i-th pattern has N_i end and tree points and that M of these points are identical to location and type among the input object and the i-th pattern. That means that in both descriptions $N_I + N_i$ discrete end and tree points exist and that $N_I + N_i - 2M$ of them are different between the two descriptions. The penalty value is 0.5/($N_I + N_i$) for each difference; therefore for two identical descriptions the assignment is 0.5 and for two completely different descriptions the assignment is 0.

The other half of the initial maximum basic probability assignment (i.e. 0.5) is assigned to a pattern that has exactly the same type, locations and orientations of subpatterns as the input object. Each difference at the location of the subpatterns decreases the amount of the initial basic probability assignment by a certain penalty value. Each difference at the type and the orientation of the subpatterns decreases the initial basic probability assignment by a smaller penalty value, in the case of the same subpatterns location. Following the analysis of the end and tree points, the penalty value is 0.5/($N_I + N_i$) for each difference at the locations of the subpatterns. The penalty value is smaller for differences at the type and the orientation of subpatterns that have the same location, in order to increase the immunity of the system against noise that does not alter the topological description but influence to a subpattern characterization only. Therefore the penalty value in this case is 0.2/($N_I + N_i$).

The initial basic probability assignment of a pattern is the summation of the initial basic probability resulted from the end and tree points and of the initial basic probability assignment resulted from the subpatterns. Each basic probability assignment results from the division of each initial basic probability assignment by the sum of the initial basic probability assignments in order to satisfy (10).

Classification of the object.

The Dempster's rule of combination (13) is used for the calculation of the basic probability assignments and the classification of the object to a certain class. The structural information is critical and invaluable to the whole learning and recognition process, because of its ability to discriminate among similar patterns that differ in small holes or connected components. From another point of view the statistical information provided by the moments is also important for the discrimination of different patterns with similar structure.

REFERENCES

[1] H. Samet, "The quadtree and related hierarchical data structures", *Computing Survey*, vol. 16, No. 2, pp. 187-260, 1984.

[2] H. Freeman, "Computer processing of line drawings", *ACM Computing Surveys*, vol. 6, pp. 57-97, 1974.

[3] D.W. Paglieroni and A.K. Jain, "Control point transfroms for shape representation and measurement", *Computer Vision, Graphics and Image Processing,* vol. 42, pp. 87-111, 1988.

[4] R.L. Kashyap and R. Chellappa, "Stochastic models for closed boundary analysis: Representation and reconstruction", *IEEE Trans. Information Theory,* vol. IT-27, pp. 627-637, No. 5, September 1981.

[5] J. Piper, "Efficient implementation of skeletonisation using interval coding", *Pattern Recognition Letters,* vol. 3, pp. 389-397, 1985.

[6] B.G. Mertzios, "Block realization of 2-D IIR digital filters", *Signal Processing*, vol. 7, No. 2, pp. 135-149, October 1984.

[7] B.G. Mertzios, "Fast block implementation of two-dimensional recursive digital filters via VLSI array processors", *Archiv fur Elektronik und Ubertragungstechnik (AEU),* vol. 43, No. 1, pp. 55-58, 1990.

[8] X. Liu and A. Fettweis, "Multidimensional digital filtering by using parallel algorithms based on diagonal processing", *Multidimensional Systems and Signal Processing,* vol. 1, pp. 51-66, 1990.

[9] I.M. Spiliotis and B.G. Mertzios, "Real-time computation of statistical moments on binary images using block representation", Proceedings of the 4*th International Workshop on Time-Varying Image Processing and Moving Object Recognition*, Florence, Italy, June 10-11, pp. 27-34, 1993.

[10] I.M. Spiliotis, D.A. Mitzias and B.G. Mertzios, "A skeleton-based hierarchical system for learning and recognition", *Proceedings of the International Symposium on the Mathematical Theory of Networks and Systems*, (*MTNS 93),* Regensburg, Germany, August 2-6, pp. 873-878, 1993.

[11] S.A. Dudani, K.J. Breeding and R.B. McGhee, "Aircraft identification by moment invariants", *IEEE Trans. Comput.*, vol. C-26, pp. 39-45, Jan. 1977.

[12] G.L. Cash and M. Hatamian, "Optical character recognition by the method of moments", *Computer Vision, Graphics and Image Processing,* vol. 39, pp. 291-310, 1987.

[13] K. Tsirikolias and B.G. Mertzios, "Statistical pattern recognition using efficient 2-D moments with applications to character recognition", *Pattern Recognition*, vol. 26, No. 6, pp. 877-882, 1993.

[14] F.A. Sadjaji and E.L. Hall, "Three-dimensional moments invariants", *IEEE Trans. Pattern Anal., Machine Intel.*, vol. PAMI-2, No. 2, pp. 127-136, 1980.

[15] A.P. Reeves, M.L. Aleey and O.R. Mitchell, "A moment based two-dimensional edge operator", *Proc. IEEE Conf. Comput. Vision, Pattern Recogn.*, pp. 114-120, 1983.

[16] A. Khotanzad and Y.H. Hong, "Invariant image recognition by Zernike moments", *IEEE Trans. Pattern Anal., Machine Intel.*, vol. 12, No. 5, pp. 489-497, May 1990.

[17] M.-K. Hu, "Pattern recognition by moments invariants", *Proc. IRE*, vol. 49, Sept. 1961.

[18] M.-K. Hu, "Visual pattern recognition by moment invariants", *IRE Trans. Inform. Theory*, vol. IT-8, pp. 179-187, Feb. 1962.

[19] M.R. Teague, "Image analysis via the general theory of moments", *J. Opt. Soc. Amer.*, vol. 70, pp. 920-930, Aug. 1980.

[20] Y.S. Abu-Mostafa and D. Psaltis, "Image normalization by complex moments", *IEEE Trans. Pattern Anal., Machine Intel.*, vol. PAMI-7, No. 1, pp. 46-55, Jan. 1985.

[21] C.-H. Teh and R.T. Chin, "On image analysis by the methods of moments", *IEEE Trans. Pattern Anal., Machine Intel.*, vol. 10, No. 4, pp. 496-513, 1980.

[22] A. Papoulis, *Propability, Random Variables, and Stochastic Processes*, McGraw-Hill, New York, 1965.

[23] G.Y. Tang, "A discrete version of Green's theorem", *IEEE Trans. Pattern Anal., Machine Intel.,* vol. PAMI-4, No. 3, May 1982.

[24] X.Y. Jiang and H. Bunke, "Simple and fast computation of moments", *Pattern Recognition,* vol. 24, No. 8, pp. 801-806, 1991.

[25] B.C. Li and J. Shen, "Fast computation of moments invariants", *Pattern Recognition,* vol. 24, No. 8, pp. 807-813, 1991.

[26] M. Hatamian, "A real-time two-dimensional moment generating algorithm and its single chip implementation", *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. ASSP-34, No. 3, pp. 546-553, June 1986.

[27] H. Blum, "A transformation for extracting new descriptors on shape", *Symposium on Models for Perception of Speech and Visual Form*, Weiant Whaten-Dunn, Ed., MIT Press, Cambridge, Mass., 1967.

[28] C.J. Hilditch, "Linear skeletons from square cupboards", *Machine Intel.*, vol. 4, pp. 403-420, 1969.

[29] L. Ji and J. Piper, "Fast homotopy - preserving skeletons using mathematical morphology", *IEEE Trans. on Pattern Anal., Machine Intel.*, vol. 14, No. 6, June 1992.

[30] A. Y. Wu, S. K. Bhaskar and A. Rosenfeld, "Parallel computation of geometric properties from the Medial Axis Transform", *Computer Vision, Graphics and Image Proc.*, vol. 41, pp. 323-332, 1988.

[31] Y. Xia, "Skeletonization via the realization of the fire's front propagation and extinction in digital binary shapes", *IEEE Trans. on Pattern Anal., Machine Intel.*, vol. 11, No. 10, pp. 1076-1086, 1989.

[32] O. Baruch, "Line thinning by line following", *Pattern Recognition Letters*, vol. 8, pp. 271-276, 1988.

[33] R. M. K. Sinha, "A width independent algorithm for character skeleton estimation", *Computer Vision, Graphics and Image Proc.*, vol. 40, pp. 388-397, 1987.

[34] L. Lam, S. W. Lee and C. Y. Suen, "Thinning methodologies - a comprehensive survey", *IEEE Trans. on Pattern Anal., Machine Intel.*, vol. 14, No. 8, pp. 869-885, 1992.

[35] A.P. Dempster, "Upper and lower propabilities induced by a multivalued mapping", *Annals of Mathematical Statistics*, vol. 38, pp. 325-339, 1967.

[36] G. Shafer, *A Mathematical Theory of Evidence*, New Jersey: Princeton University Press, 1976.

[37] R. Hummel and M. Landy, "A statistical viewpoint on the theory of evidence", *IEEE Trans. on Pattern Anal., Machine Intell.*, vol. 10, pp.235-247, March 1988.

[38] T. Start, "Continuous belief functions for evidential reasoning", *Proc. AAAI-84,* Austin TX, pp. 896-903, 1984.

[39] E.L. Waltz and D.M. Buede, "Data fusion and decision support for command and control", *IEEE Trans. on Systems, Man, Cyber.*, vol. SMC-16, no. 6, pp. 865-879, Nov./Dec. 1986.

[40] P.L. Bogler, "Shafer-Dempster reasoning with applications to multisensor target identification systems", *IEEE Trans. on Systems, Man, Cyber.*, vol. SMC-17, no. 6, pp. 968-977, Nov./Dec/ 1987.

[41] J.R. Bourne et al., "Organizing and understanding beliefs in advice-giving diagnostic systems", *IEEE Trans. on Knowledge and Data Engin.*, vol. 3, no. 3, pp. 269-280, September 1991.

[42] D. Dubois and H. Prade, "Fuzzy sets and statistical data", *Europ. Jour. of Operational Research*, vol. 25, pp. 345-356, 1986.

[43] S.J. Henkind and M.C. Harrison, "An analysis of four uncertainty calculi", *IEEE Trans. on Systems, Man, Cyber.*, vol. 18, no. 5, pp. 700-714, Sep./Oct. 1988.

[44] I.M. Spiliotis and B.G. Mertzios, "Processing and analysis on binary images using image block representation". Submitted to *Pattern Recognition.*